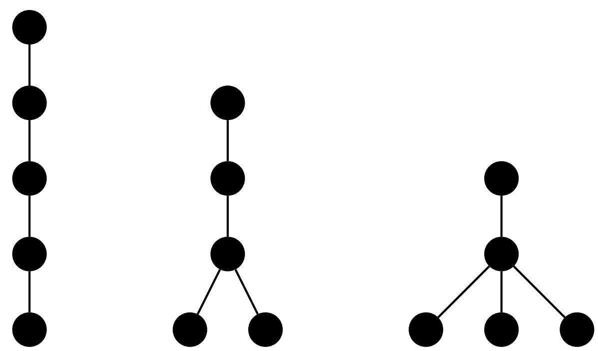
1 Trees

Recall that we finished the last semester with the introduction of connectedness and connected components. This leads us to an interesting and useful class of graphs with several interesting properties.

Definition 1. A connected graph G which does not contain any cycles is a *tree*.

Among other things, this explains why we call an acyclic graph a forest — it's a graph whose components are trees! Below the three distinct trees on five nodes are pictured: note that they look somewhat "tree-like", visually speaking:



We will prove an important property of trees, which will give us an extremely useful tool for building inductive proofs about trees.

Proposition 1. *If T is a tree with at least 2 vertices, then $\delta(T) = 1$*

Proof. Note that since $|T| \geq 2$, every vertex in T has degree of at least 1, since a vertex of degree zero would not be connected to other vertices. We shall show that not every vertex can not have degree 2.

Suppose that $\delta(T) \geq 2$. Then, choose a vertex u_0 in T. Since $d(u_0) \geq 1$, u_0 has a neighbor u_1. Since $d(u_1) \geq 2$, u_1 has at least two neighbors: u_0 and some vertex $u_2 \neq u_0$. Likewise, u_2 has u_1 and some distinct u_3 as neighbors. We may continue this process for as long as desired to get a walk $u_0 \sim u_1 \sim u_2 \sim u_3 \sim \cdots \sim u_k$ such that every three consecutive vertices are distinct for arbitrarily large k. By making k sufficiently large, since T is finite, this walk is guaranteed to self-intersect. Let j be the first index of self-intersection, so that for some $i < j$, $u_i = u_j$. By our distinctness criterion, $i \neq j-1$ and $i \neq j-2$, so $i \leq j-3$. Then, $u_i \sim u_{i+1} \sim \cdots \sim u_{j-1}$ is a path of length at least 2, and thus $u_i \sim u_{i+1} \sim \cdots \sim u_{j-1} \sim u_j = u_i$ is a cycle; since T cannot contain a cycle, our assumption that $\delta(T) \geq 2$ must be erroneous. □

Definition 2. A vertex of degree 1 in a tree is called a *leaf*.

The above proposition, then, shows that every tree has at least one leaf. It's in fact fairly each to modify the proposition to show that every tree has at least two leaves, but not entirely necessary.

One useful property of leaves is that they can be pruned:

Theorem 1. *If v is a leaf of tree T, then $T - v$ is also a tree.*

Proof. $T - v$ is trivially acyclic: removing elements from a graph without a cycle will result in a graph which is still acyclic. It is less obvious that $T - v$ is connected, however. Consider vertices

$u, w \in T - v$. By connectivity of T, there is a path from u to w in T: $u = v_0 \sim v_1 \sim \cdots \sim v_k = w$. Since for $0 < i < k$, each v_i is adjacent to the distinct vertices v_{i-1} and v_{i+1}, $d_T(v_i) \geq 2$ so $v_i \neq v$. Likewise, since u and w are in $T - v$, they are not equal to v either. Thus, the aforementioned path lies in $T - v$, so $T - v$ is connected. □

If we reverse this pruning process, we get the property that *any* tree can be built by gluing leaves onto a basic tree.

Theorem 2 (Structure Theorem for Trees)**.** *For any tree T with $|T| = n$, there is a sequence of subtrees $T_1 \subset T_2 \subset \cdots \subset T_n = T$ such that:*

- *T_1 is the trivial one-vertex tree.*
- *For $i > 1$, $V(T_i) = V(T_{i-1}) \cup \{v_i\}$ and $E(T_i) = E(T_{i-1}) \cup \{\{v_i, u\}\}$ for some vertex u in T_{i-1}.*

Proof. We shall prove this by induction on n. For $n = 1$, the sequence is trivially $T_1 = T$.

For the inductive step, given $|T| = n$, let v_n be a leaf of T. By the proposition above, $T_{n-1} = T - v_n$ is a tree; the inductive hypothesis tells us that there is a sequence $T_1 \subset T_2 \subset \cdots \subset T_{n-1}$ satisfying the conditions above. Now, by the construction of T_{n-1}, $V(T) = V(T_{n-1}) \cup \{v_n\}$, and the edges of T are the edges of T_{n-1} together with the lone edge incident to v_n. □

The above may alternatively be restated as a claim that there is an ordering $(v_1, v_2, v_3, \ldots, v_n)$ of the vertices of T such that for each k, the induced subgraph $T[v_1, v_2, \ldots, v_k]$ is also a tree.

This is known as a *structural theorem* because it gives a procedure for building trees step by step. Structural theorems are excellent for proving properties by induction, since one can keep track of properties introduced at each stage of the construction. For instance:

Proposition 2. *If a tree T has n vertices, then it has $n - 1$ edges.*

Proof. Every tree on n vertices can be constructed by $n - 1$ "vertex-gluings" to the trivial tree T_1. T_1 has one vertex and no edges; each gluing adds one edge and one vertex by construction, so $\|T\| = 0 + (n-1) \cdot 1 = n - 1$. □

One last cute property of trees:

Proposition 3. *For any two vertices u and v in a tree T, there is a unique path from u to v.*

Proof. T is connected, so there is by definition at least one path from u to v. We shall show that there are not two distinct paths from u to v.

Suppose there are two distinct paths $u = u_0 \sim u_1 \sim u_2 \sim \cdots \sim u_k = v$ and $u = u'_0 \sim u'_1 \sim u'_2 \sim \cdots \sim u'_\ell = v$. We know that $u_0 = u'_0$, but since the paths are distinct, there must be some value i such that $u_i \neq u'_i$. Let us consider the first such u_i: then $u_{i-1} = u'_{i-1}$ but $u_i \neq u'_i$. Now let us consider the path $u_i \sim u_{i+1} \sim u_{i+2} \sim \cdots \sim u_k = v$. This path must intersect $u'_i \sim u'_{i+1} \sim u'_{i+2} \sim \cdots \sim u'_\ell = v$ at least once, since v is in both paths; let j be the least index at which $u_j = u'_{j'}$ for some j'.

Now, we can construct an explicit cycle as such:

$$u_{i-1} \sim u_i \sim u_{i+1} \sim \cdots \sim u_j = u'_{j'} \sim u'_{j'-1} \sim u'_{j'-2} \sim \cdots \sim u'_i \sim u'_{i-1}$$

We can show this is a cycle by showing that every vertex visited is distinct. Since it is a path, $u_{i-1} \sim \cdots u_j$ consists of distinct vertices, and by the choice of j, all of its vertices except u_j are distinct from the u' vertices; in addition, since $u'_{i-1} \sim \cdots u'_{j'}$ is a path, all the u' vertices are distinct from each other.

We have thus shown that two distinct paths between u and v necessitate a cycle in T, which is impossible, so there is no more than one path between u and v. □

The above argument is also amenable to a structural-theorem based proof, presented here for contrast.

Proof. We perform an inductive proof on $|T|$. When $|T| = 1$, there is only one path from the unique vertex of T to itself, so the proposition is vacuously true. Now, let us prove the inductive step.

For $|T| = n$, we know by the tree-structural results that $T = T' + v_n + e$, where v_n is a new vertex and e is an edge $\{u, v_n\}$, for some $u \in V(T')$. We know by the inductive hypothesis that paths in T' are unique; Now we need to show that paths in T are also unique. Every path in T' is a path in T, so we need to show two things:

- For $v, w \in V(T')$, there is no path from v to w using the new edge e.
- For $v \in V(T')$, there is exactly one path from v to v_n in T.

Both of these are fairly straightforward. Any path visiting v_n must terminate at v_n, since there is only one incident edge to v_n; thus the first statement is true, and the second follows from noting that any path from $v \in V(T')$ to v_n has final two vertices $u \sim v_n$, so every path from v to v_n is an extension of a path from v to u. □

2 Spanning trees and connected structure

Definition 3. A subgraph of a graph G which is a tree with the same vertex-set as G is called a *spanning tree.*

Using a Lemma which you're asked to prove, we can show that every connected graph does in fact have a spanning tree.

Lemma 1. *If graph G is connected and contains a cycle, then there is an edge e in G such that $G - e$ is still connected.*

Proof. Left as an exercise for the reader (literally: it's question #1 on the problem set). □

Theorem 3. *Every connected graph G has a spanning tree.*

Proof. Let $G_0 = G$. G_0 is a connected graph on the vertex-set of G; if G_0 contains no cycles, it is a tree and thus is the spanning tree of G. If G_0 contains a cycle, then by the above lemma, there is an edge in G_0 such that $G_0 - e$ is connected. Let $G_1 = G_0 = e$.

G_1, like G_0, is a connected graph on the vertex-set of G. If it's a tree, we're done; if it's not a tree, then there is an edge e such that $G_2 = G_1 - e$ is connected.

We continue in suchlike fashion, building a chain $G_0 \supset G_1 \supset G_2 \supset \cdots$ of graphs formed by removing edges. Since there are a finite number of edges to start with, this process must terminate; as seen above, it terminates when G_n contains no cycles. The graph formed by doing so is a tree on G's vertex set using edges from G, and is thus a spanning tree for G. □

If you don't like the "arbitrary-length descent" argument above, it is easy to reformulate it as an inductive argument on $\|G\|$.

One nice thing about a spanning tree is that we can use the structural theorem on it to build structure for the connected graph it's in:

Proposition 4. *For any connected graph G, there is an ordering of its vertices $v_1, v_2, \ldots, v_n$ such that for each k, the induced graph $G[v_1, v_2, \ldots, v_n]$ is connected.*

Proof. Let T be a spanning tree of G. By the structural theorem on trees, there is an ordering of the vertices such that $T[v_1, v_2, \ldots, v_k]$ is a tree for each k. Since T is a subgraph of G, every edge in T appears in G, so $T[v_1, v_2, \ldots, v_k] \subset G[v_1, v_2, \ldots, v_k]$. Thus every path in $T[v_1, \ldots, v_k]$ appears in $G[v_1, \ldots, v_k]$, so by connectivity of trees, $G[v_1, \ldots, v_k]$ is also connected. □

We can in fact convert this into a structural theorem closely akin to our structural theorem for trees:

Theorem 4 (Structural theorem on connected graphs)**.** *For any connected graph G with $|G| = n$, there is a sequence of connected subgraphs $G_1 \subset G_2 \subset \cdots \subset G_n = G$ such that:*

- *G_1 is the trivial one-vertex graph.*
- *For $i > 1$, $V(G_i) = V(G_{i-1}) \cup \{v_i\}$ and $E(G_i) = E(G_{i-1}) \cup \{\{v_i, u_1\}, \{v_i, u_2\}, \ldots, \{v_i, u_{k_i}\}$ for some nonempty set of vertices $u_1, u_2, \ldots, u_{k_i}$ in G_{i-1}.*

Proof. Let $G_i = G[v_1, v_2, \ldots, v_i]$ with the vertex-enumeration established in the previous proposition, and this can be shown to satisfy the above criteria. □